Проект по Разпределени софтуерни архитектури

**Изобразяване на фрактал**

Изготвил: Виктор Христов, СИ 3, №62151

Дата: 13.6.2020

# Увод

# Цел и предназначение на проектираното приложение

Целта на този проект е да се създаде програма, която генерира изображение на множество на Манделброт с функция **Zn+1=C∗cos(Zn )** за комплексно число C чрез реализиране на паралелен алгоритъм.

Множеството на Манделброт е вид фрактал. Това са фигури, които рекурсивно съдържат себеподобни фигури по своите граници. Всеки фрактал се изобразява чрез множество от комплекси числа. Комплексните числа се пресмятат чрез използване на комплексна равнина. Това е координатна система, при която реалната част на числото е абциса, а имагинерната - ордината. След като се знае стойността на комплексното число, то тогава може да се определи дали то принадлежи към конкретния фрактал. Прилага се следната система:

1. Z0=0
2. Zn+1=C∗cos(Zn )

Ако след избран от програмиста брой итерации на функцията не може да се определи дали комплексното число клони към безкрайност, то тогава се приема, че принадлежи към това множество на Манделброт.

За да се генерира пълното изображение, ще е нужно да се изчисли всяка точка от зададената комплексна равнина, независимо дали е част от множеството на Манделброт или не. Единствено след като е приключило изчислението ще може да се запази генерираното изображение във визуален формат.

# Функционален анализ

В тази подточка ще се анализират примерни решения на задачата за изобразяване на множество на Манделброт. Всяка от тях обяснява какво представлява задачата, как чрез комплекси числа се изобразява на координатна система и как се пресмята дали едно число е част от множеството. Поради това при анализа ще се обърне внимание само на по-значителните части от статиите, като най-важната от тях е реализацията на паралелната обработка.

1. Parallel Implementation and Analysis of Mandelbrot Set Construction [[1]](#_Списък_източници)

Авторът обръща внимание на проблема, че при разделянето на изображението на големи задания се получа лошо балансиране на работата на процесите, и предлага своето решение.

Идеята на алгоритъма е всеки процес да изчислява произволно избрани пиксели от изображението. Инициализират се два вектора, които представляват интервала на координатите X и Y. Произволно се генерира число, спрямо което стойностите на векторите се разбъркват. След това векторите се използват, за да се генерира изображението.

Комуникацията между процесите е минимизирана. Преди да се извършат изчисленията, процесите получават информация за векторите, след което използват тази информация, за да пресметнат елементите, които са им разпределени. Накрая главния процес събира всичките задания и сглобява пълното изображение.

1. Parallel Fractal Image Generation - A Study of Generating Sequential Data with Parallel Algorithms [[2]](#_Списък_източници)

Авторът на тази статия споменава важността от броя на итерациите при изчисляване дали някоя точка принадлежи на множеството на Манделброт. Някои точки се нуждаят от огромен брой итерации, за да определи дали числото клони към безкрайност.

Колкото повече итерации се извършват, толкова по-правилно ще е крайното изображение. Но от друга гледна точка, колкото повече итерации се извършват, толкова по-бавно ще се генерира изображението. Поради това е важно да има компромис между точността и бързодействието на алгоритъма.

1. Parallel Mandelbrot in Julia, C++, and OpenCL [[3]](#_Списък_източници)
2. Parallel Processing and The Mandelbrot Set [[4]](#_Списък_източници)

# Нефункционален анализ

1. Технологии
2. Модел на обслужването
3. Софтуерен модел
4. Коментар и обосновка на избраното решение

# Проектиране

# Функционално проектиране

# Нефункционално проектиране

# Тестване

# Списък източници

[1] Isaac K. Gäng, David Dobson, Jean Gourd and Dia Ali,

Parallel Implementation and Analysis of Mandelbrot Set Construction, <https://www.academia.edu/1399383/Parallel_Implementation_and_Analysis_of_Mandelbrot_Set_Construction>

[2] Matthias Book,

Parallel Fractal Image Generation - A Study of Generating Sequential Data with Parallel Algorithms,

presented May 3, 2001,

<http://matthiasbook.de/papers/parallelfractals/index.html>

[3] Distrust Simplicity,

Parallel Mandelbrot in Julia, C++, and OpenCL, January 9, 2015,

<http://distrustsimplicity.net/articles/mandelbrot-speed-comparison/>

[4] Carmen Pughineanu,

Parallel Processing and The Mandelbrot Set,

<http://www.afahc.ro/ro/revista/Nr_2_2008/ART_CARMEN.pdf>